CLAIMS

1. A method for executing a signed applet packaged in a given file, comprising:

upon loading a class, determining whether a signature in the given file type applies to the class;

if so, executing a verification procedure to verify the signature and the identity of a signer that generated the signature;

following a successful verification, determining whether the signer is identified in a policy entry; and

if the signer is identified in the policy entry, populating a permission set for the class.

2. The method as described in Claim 1 wherein the signature is verified using a given algorithm used to sign the applet.

3. The method as described in Claim 2 wherein the given algorithm is selected from the set of algorithms consisting of DSA/SHA1, RSA/MD5 and RSA/SHA1.

4. The method as described in Claim 1 wherein the step of populating the permission set for the class awards the class a permission as specified in the policy entry.

5. The method as described in Claim 1 further including the steps of:

determining whether the applet has made a request that requires permission; and

if so, using the permission set of the class to determine whether the class has the permission.

6. The method as described in Claim 5 further including the step of:

responding to the request if the class has the permission.

7. The method as described in Claim 1 wherein the step of verifying the identity of the signer verifies that the signer is in a default certificate database and that a certificate of the signer has not expired.

8. The method as described in Claim 1 wherein the step of verifying the identity of the signer verifies that the signer contains a certificate chain to a trusted certificate authority, that each certificate in the certificate chain contains a signature that can be verified by a given key, and that each certificate in the certification chain has not expired.

9. The method as described in Claim 1 wherein the given file is selected from the set of file types consisting of a first signed jar file, a second signed jar file, and a signed cab file.

10. The method as described in Claim 1 wherein the signed applet is executable in a given one of a set of different browser types.

11. A method for executing a signed applet packaged in a given file, comprising:

upon loading each class, determining whether any signatures in the given file applies to the class;

if so, executing a verification procedure to verify the signature and the identity of a signer that generated the signature;

following a successful verification, determining whether the signer is identified in a policy entry;

if the signer is identified in the policy entry, awarding the class a permission as identified in the policy entry; and

responsive to a request that requires a permission, using the permission set for the class to determine whether the class has the permission.

12. The method as described in Claim 11 further including the step of responding to the request if the class has the permission.

13. The method as described in Claim 11 wherein the step of verifying the identity of the signer verifies that the signer is in a default certificate database and that a certificate of the signer has not expired.

14. The method as described in Claim 11 wherein the step of verifying the identity of the signer verifies that the signer contains a certificate chain to a trusted certificate authority, that each certificate certificate in the certificate chain contains a signature that can be verified by a given key, and that each certificate in the certification chain has not expired.

15. The method as described in Claim 11 wherein the given file is selected from the set of file types consisting of a first signed jar file, a second signed jar file, and a signed cab file.

16. The method as described in Claim 11 wherein the signed applet is executable in a given one of a set of different browser types.

17. A computer program product including computer usable code for use in a Java runtime environment (JRE), comprising:

an applet class loader for loading a set of applet classes archived in a signed file;

a set of signature engine classes for verifying applet class signatures; and

a security manager class callable by the applet class loader upon receipt of an initial request that requires a given permission and, in response thereto invoking a policy file class that verifies a signer based on the existence of a matching certificate in a set of keystores.

18. The computer program product as described in Claim 17 wherein the set of signature engine classes includes a DSA/SHA1 class, an RSA/MD5 class, and a RSA/SHA1 class.

19. The computer program product as described Claim 17 wherein the applet class loader is invoked by a Java Plug-in of the Java runtime environment.

20. The computer program product as described in Claim 17 wherein the applet classes are archived in a jar file.

21. The computer program product as described in Claim 17 wherein the applet classes are archived in a cab file.

22. A system, comprising:

a browser;

a Java runtime environment;

a set of keystores;

an applet class loader for loading a set of applet classes archived in a signed file;

a set of signature engine classes for verifying applet class signatures; and

a security manager class callable by the applet class loader upon receipt of an initial request that requires a given permission and, in response thereto, invoking a policy file class that verifies a signer based on the existence of a matching certificate in the set of keystores.

23. The system as described in Claim 22 wherein at least one signature engine verifies signatures using a given algorithm used to sign the applet.

24. The system as described in Claim 22 further comprising means for populating a permission set for the class, wherein the class is awarded a permission as specified in a policy entry in a database managed by the security manager class.

25. The system as described in Claim 22 wherein the signed file is selected from the set of file types consisting of a first signed jar file, a second signed jar file, and a signed cab file.